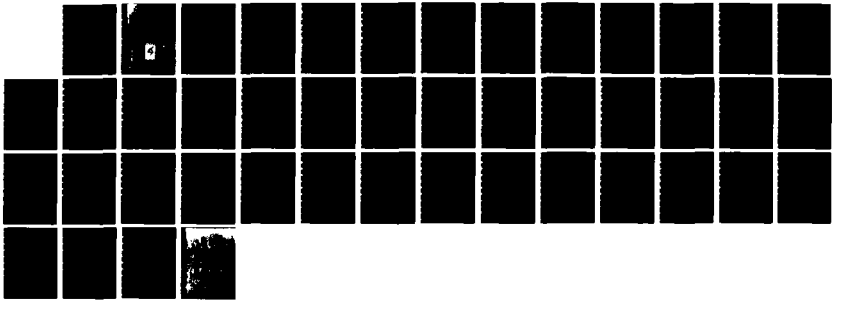
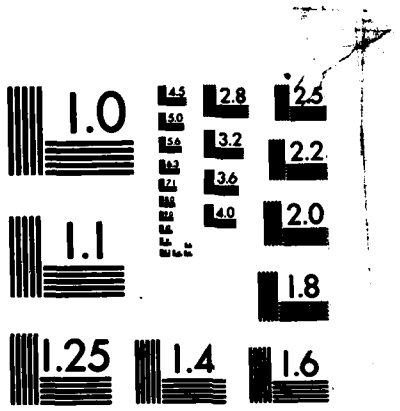
AD-A140 877 HARMONY THEORY: PROBLEM SOLVING PARALLEL COGNITIVE MODELS AND THERMAL PHY..(U) CALIFORNIA UNIV SAN DIEGO LA JOLLA INST FOR COGNITIVE SCIENCE.. 1/1

UNCLASSIFIED P SMOLENSKY ET AL. APR 84 ICS-8403 F/G 12/1 NL





(12)

# HARMONY THEORY:
## Problem Solving, Parallel Cognitive Models, and Thermal Physics

Paul Smolensky and Mary S. Riley




INSTITUTE FOR COGNITIVE SCIENCE
UNIVERSITY OF CALIFORNIA, SAN DIEGO LA JOLLA, CALIFORNIA 92093

ICS Report 8404
April 1984

# HARMONY THEORY:
## Problem Solving, Parallel Cognitive Models, and Thermal Physics

Paul Smolensky and Mary S. Riley

*Institute for Cognitive Science*
*University of California, San Diego*
*La Jolla, California 92093*



The first two papers in this report will appear in the
*Proceedings of the Sixth Annual Meeting of the Cognitive Science Society.*
Boulder, Colorado, June 1984.

*The research reported here was conducted under Contract N00014-79-C-0323, NR 667-437 with the Personnel and Training Research Programs of the Office of Naval Research, and was sponsored by the Office of Naval Research and a grant from the System Development Foundation. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the sponsoring agencies. Approved for public release; distribution unlimited. Reproduction in whole or in part is permitted for any purpose of the United States Government.*

*ONR REPORT 8403*

Requests for reprints should be sent to the authors at Institute for Cognitive Science, C-015; University of California, San Diego; La Jolla, California, 92093, USA.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|---|---|
| 1. REPORT NUMBER ONR #8403 | 2. GOVT ACCESSION NO. AD-A140 877 | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle) Harmony Theory: Problem Solving, Parallel Cognitive Models, and Thermal Physics | | 5. TYPE OF REPORT & PERIOD COVERED Technical Report |
| | | 6. PERFORMING ORG. REPORT NUMBER |
| 7. AUTHOR(s) Paul Smolensky and Mary S. Riley | | 8. CONTRACT OR GRANT NUMBER(s) N00014-79-C-0323 |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS Center for Human Information Processing Institute for Cognitive Science University of California, San Diego La Jolla, CA 92093 | | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NR 667-437 |
| 11. CONTROLLING OFFICE NAME AND ADDRESS Personnel & Training (Code 442PT) Office of Naval Research 800 No. Quincy St., Arlington, VA 22217 | | 12. REPORT DATE April, 1984 |
| | | 13. NUMBER OF PAGES |
| 14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) | | 15. SECURITY CLASS. (of this report) Unclassified |
| | | 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

This research was also supported by a grant from the System Development Foundation.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

| | |
|---|---|
| production systems | intuitive physics |
| parallel distributed processing | harmony theory |
| thermal cognitive models | statistical physics |
| problem solving | schema theory |

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

OVER

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE S/N 0102 LF 014 6601

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

## ABSTRACT

**In the first of three papers in this report, we describe a parallel model designed to solve a class of relatively simple problems from elementary physics, and discuss the implications for models of problem solving in general. We show how one of the most salient features of problem solving, sequentiality, can *emerge naturally* within a parallel model that has no explicit knowledge of how to sequence analysis. This model exploits a new type of parallel distributed processing that employs stochastic processors and is based on a formal mapping between parallel computation and thermal physics. The mathematical theory of this type of processing—harmony theory—is discussed in the second and third papers.**

# HARMONY THEORY:

# PROBLEM SOLVING, PARALLEL COGNITIVE MODELS, AND THERMAL PHYSICS

## Contents



The authors discuss

In the first paper, we describe a parallel model designed to solve a class of relatively simple problems from elementary physics, and discuss the implications for models of problem solving in general. We show how one of the most salient features of problem solving, sequentiality, can *emerge naturally* within a parallel model that has no explicit knowledge of how to sequence analysis. This model exploits a new type of parallel distributed processing that employs stochastic processors and is based on a formal mapping between parallel computation and thermal physics. The mathematical theory of this type of processing—harmony theory—is discussed in the second and third papers.

# A PARALLEL MODEL OF (SEQUENTIAL) PROBLEM SOLVING

**Mary S. Riley and Paul Smolensky**

*Institute for Cognitive Science*
*University of California, San Diego*
*La Jolla, California 92093*

**April, 1984**

**Abstract**

**We contrast the production system and parallel distributed processing approaches to modelling simple electric circuit problem-solving. We show how sequentiality can emerge naturally within a parallel model that has no explicit knowledge of how to sequence analysis.**

**To appear in the**
***Proceedings of the Sixth Annual Meeting of the Cognitive Science Society.***

---

**This research was supported by a grant from the System Development Foundation and by contract N00014-79-C-0323, NR667-437 with the Personnel and Training Research Programs of the Office of Naval Research.**

# A PARALLEL MODEL OF (SEQUENTIAL) PROBLEM SOLVING

## Nature of Rules and Their Interaction

This paper is concerned with the nature of the rules involved in solving problems and the interaction among those rules. We describe a parallel model designed to solve a class of relatively simple problems from elementary physics and discuss its implications for models of problem solving in general. We show how one of the most salient features of problem solving, sequentiality, can *emerge naturally* within a parallel model that has no explicit knowledge of how to sequence analysis.

Consider the problem shown in Figure 1. The task is to determine the qualitative effects of increasing the resistance of $R_2$ on other circuit values, assuming the applied voltage and resistance of $R_1$ remain unchanged.

A common approach to modelling the process of solving problems like these is to assume that knowledge is organized as a production system, similar to that shown in Table 1 (see Riley, 1984, for a review). Here the model's rules for making inferences are in the form of condition-action pairs, or *productions*. The condition specifies the particular elements and relations that must be present in the data base in order for the condition to be true. When the production system is solving a problem, the conditions of the various productions are tested in order until one of them is true; the action of that production is then performed. The action generally makes some change in the data base which in turn means the condition of a different production will be true, causing another action to be performed.

Since production systems are universal computers, they can be programmed to display any behavior (Newell, 1981). However certain kinds of behavior can be achieved with other styles of computation in more economical, elegant, extendible and natural ways. Features that are intrinsic to, or naturally incorporated within, a pure production system approach are:

1) *Sequentiality:* each action taken utilizes the knowledge contained in precisely one rule.
2) *Directionality:* the knowledge encoded in each rule has a distinct directionality from input (condition) to output (action).
3) *Exact matching:* each rule acts only when a perfect match to its condition occurs.
4) *Determinism:* performance will be identical on all solutions of a given problem.

Within the production system approach it is difficult to naturally avoid certain difficulties:

1) *Lack of robustness under degradation of rules* (either removal of correct rules or addition of incorrect ones).
2) *Lack of robustness under ill-formed problems* that contain inconsistent or insufficient given information.
3) *Lack of variability* in routes to correct answers or in correctness of answers; a problem for modelling human behavior.
4) *Need for explicit conflict resolution rules* that determine which rule will apply when several have true conditions.

The parallel distributed processing approach represented by our model naturally avoids these

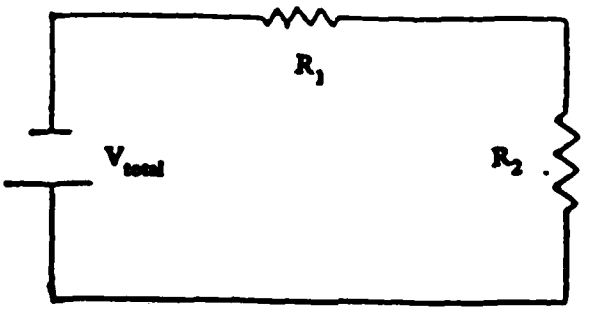



**Figure 1. A series circuit with two resistors, $R_1$ and $R_2$. What are the effects of an increase in the resistance of $R_2$, assuming that $V_{total}$ and the resistance of $R_1$ remain the same?**

## Table 1

**A Simple Production System for Solving the Problem in Figure 1.**

### Productions

| | Condition | Action |
|---|---|---|
| P1. | $<V_x$ *same*, $R_x$ *up*$>$ | $<$I *down*$>$ |
| P2. | $<R_x$ *up*, $R_y$ *same*$>$ | $<R_{total}$ *up*$>$ |
| P3. | $<V_x$ *down*, $V_{total}$ *same*$>$ | $<V_y$ *up*$>$ |
| P4. | $<R_x$ *same*, I *down*$>$ | $<V_x$ *down*$>$ |

### Problem Solution

| Problem Representation | Matched Production | |
|---|---|---|
| Cycle | Condition | Action |
| 1. $R_2$ *up*, $R_1$ *same*, $V_{total}$ *same* | P2. $<R_2$ *up*, $R_1$ *same*$>$ | $<R_{total}$ *up*$>$ |
| 2. $R_2$ *up*, $R_1$ *same*, $V_{total}$ *same*, $R_{total}$ *up* | P1. $<V_{total}$ *same*, $R_{total}$ *up*$>$ | $<$I *down*$>$ |
| 3. $R_2$ *up*, $R_1$ *same*, $V_{total}$ *same*, $R_{total}$ *up*, $I_{total}$ *down* | P4. $<R_1$ *same*, I *down*$>$ | $<V_1$ *down*$>$ |
| 4. $R_2$ *up*, $R_1$ *same*, $V_{total}$ *same*, $R_{total}$ *up*, $I_{total}$ *down*, $V_1$ *down* | P3. $<V_1$ *down*, $V_{total}$ *same*$>$ | $<V_2$ *up*$>$ |
| 5. $R_2$ *up*, $R_1$ *same*, $V_{total}$ *same*, $R_{total}$ *up*, $I_{total}$ *down*, $V_1$ *down*, $V_2$ *up* | | |

difficulties, but has its own problems, as we shall see.

## The Model

Our model has been constructed within the framework of harmony theory (Smolensky, 1983, 1984). Rules are represented as a collection of nodes in a network, as shown in Figure 2. A typical rule is $<I\ down, V_1\ down, R_1\ same>$; this rule states that the combination of changes "voltage across $R_1$ goes down, current goes down, $R_1$ stays the same" is a consistent set (Ohm's Law). In fact, the rules consist precisely of all allowed combinations of qualitative changes in circuit variables that are consistent with each circuit law. There are 65 such instances. [1]

Unlike condition-action rules, there is no directionality associated with the variables in the harmonium rules.

In a particular problem, only some of the instances represented by harmonium rules are relevant. To represent this, each rule node has an *activation value* that can be either 1 (active) or 0 (inactive).

In addition to rule nodes, the harmonium model contains nodes for representing the problem in terms of qualitative changes in circuit variables. Some nodes have values given by the problem ($R_2\ up, R_1\ same, V_{total}\ same$). The model's answer is represented by values assigned to the remaining nodes.

As shown in Figure 2, there is a connection between an individual circuit variable node and each rule involving it; this connection is labelled by the appropriate value for that variable according to that rule.

The goal of processing is to find a set of rule nodes to activate and a set of values for circuit variables that are consistent with those rules. Search toward this goal is guided by a measure of the consistency between a set of activated rules and a set of circuit variables: this measure is called the *harmony function*. The state of highest harmony should be the correct answer to the problem.

Processing is probabilistic and constructed so that at any moment, *the higher the harmony of a state, the more probable it is*. The spread in this probability distribution is determined by a system parameter called the *computational temperature T*. Initially, all rules are inactive, the circuit variables given by the problem are assigned their values, and the remaining circuit variables are assigned random values. The temperature is set to a high value, and the stochastic search begins. Rules are activated and deactivated, circuit variable values are changed (except the given ones), and states are visited in accordance with their harmony. As the search continues, this temperature is lowered, and the system displays less and less randomness, focusing in on the states of highest harmony. After a while, the temperature becomes very low, and the search is effectively stopped: an answer has been selected.

Sequentiality of deduction seems to be completely lacking from the harmonium model, although it is a very salient feature of human problem solving. Just the same, in creating this model we expected it to display an emergent seriality. If a single circuit variable is monitored during the search, it will fluctuate randomly at first, and eventually "lock in" to a value that is very resistant to change. The temperature at which this occurs is the "freezing temperature" for that variable. We expected that different variables would have different freezing temperatures, depending on the problem situation; the

1. Thirteen each for: Kirchoff's Law, ... [illegible] ...nation $R_{total} = R_1 + R_2$, and three versions of Ohm's Law (one each for $R_1$, $R_2$, and $R_{total}$).

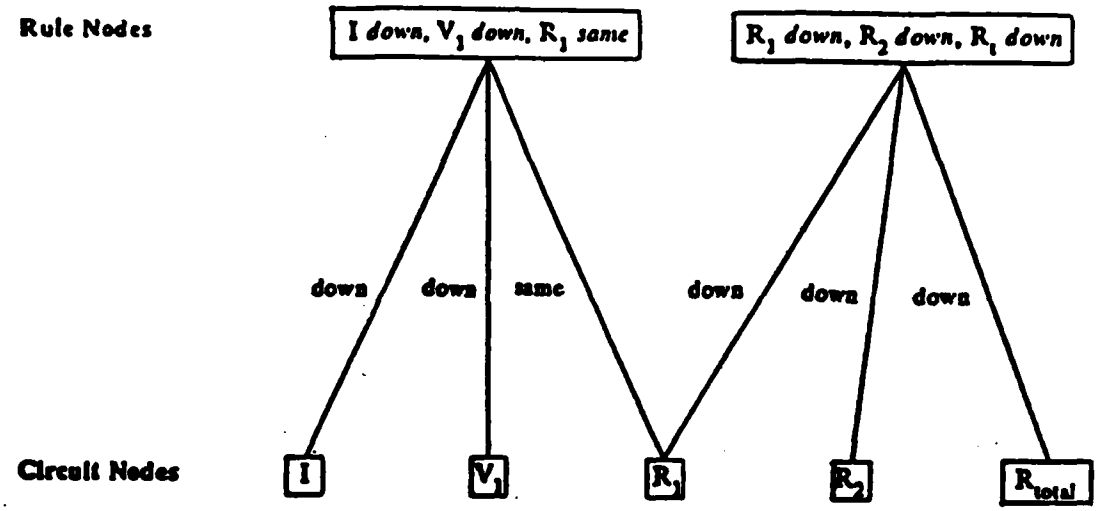



**Figure 2. A portion of the harmonium model's network.**

one with highest freezing temperature would settle first, which would in turn determine the value selected for the variable with the next highest freezing temperature, and so on.

In addition to $T$, harmonium models have a second global parameter, $\kappa$; it is the sole parameter in the definition of the harmony function. When $\kappa$ is near one, only rules that match the current guesses for circuit values *exactly* can become active without lowering the harmony of the state; for low values of $\kappa$, approximate matches are sufficient. Initially, $\kappa$ is small, approximate matching is encouraged, and many rules become activated; as the computation proceeds, $\kappa$ approaches one and the set of active rules shrinks toward the five that exactly match the answer.

As the node for each circuit variable freezes into a value, it does so under the influence of all the active rule nodes connected to it. Unlike a production system, matching for rules need not be exact, and several rules can act at the same time.

The harmony function we used, as well as the schedule for lowering $T$ and raising $\kappa$, are shown in Figure 3. A trial consisted of 400 iterations of 100 node updates each; since there are 79 nodes in the model, this corresponds to slightly over 500 updates of each node.

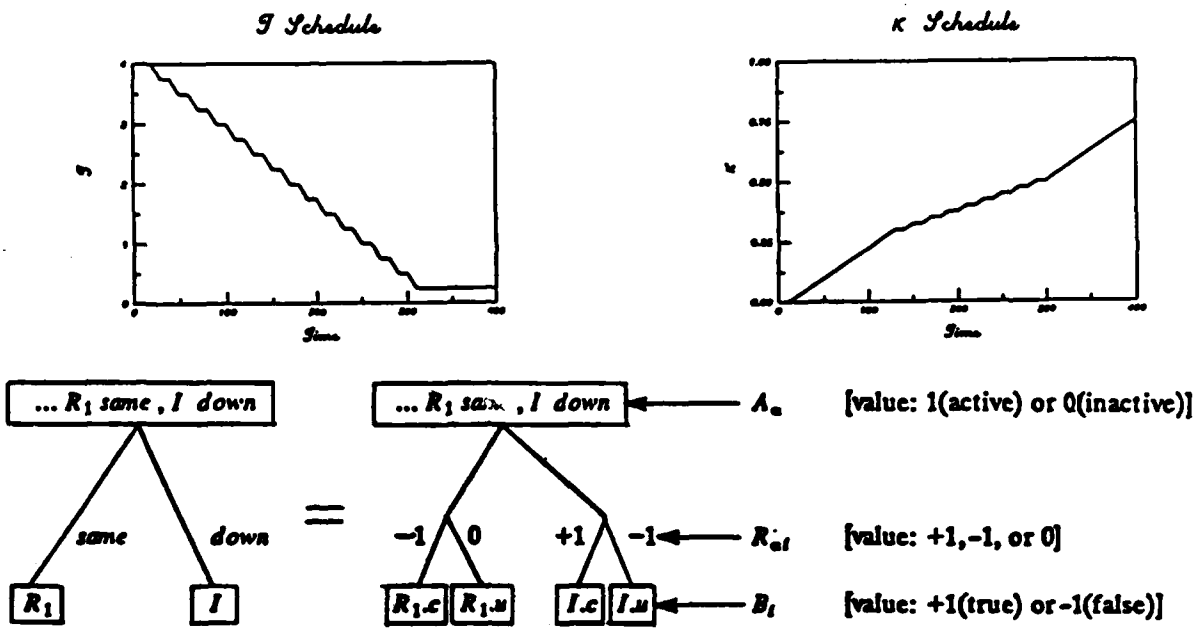
The stochasticity of the model produces variability in the behavior. In a run of 30 trials, the correct answer was produced 28 times. When the 30 values the system assigned to the circuit variables for each of the 400 iterations are averaged, Figure 4 results. In this graph, *up* is represented by 1 and *down* by −1. Initially, the values for all variables fluctuate around zero; eventually, each goes towards the correct value. The time at which the four decisions are made are indicated in the last portion of this figure, in which the region between .5 and −.5 has been removed. The sequence of assignments is $R_{total}, I_{total}, V_1, V_2$; *the sequence of "inferences" that emerges naturally from the parallel processing is exactly the same as the sequence produced in a production system model.*

The harmonium model displays both types of robustness that are difficult to achieve naturally with production systems. Since individual inferences are made under the simultaneous influence of several rules, they are less vulnerable to degradation of a single rule. When inconsistent information is given in a problem, the harmonium model finds the most consistent (highest harmony) answer possible. When insufficient information is given, the system finds one of the correct answers, and finds different answers on different trials. Such a robust tendency to form coherent interpretations of inputs is important both for modelling human cognition and for building intelligent machines.

## Extensions

While the parallel distributed processing approach has certain advantages over the production system approach, it also has grave disadvantages. The most serious is the difficulty of performing symbol manipulation. Without variable binding mechanisms, types and tokens, it is difficult to imagine how to develop a general model capable of analyzing a variety of circuits; our model is specialized to a single circuit, and even so we must replicate the rules encoding valid instances of Ohm's Law three times (once for each relevant piece of the circuit).

It may be psychologically plausible to postulate a small collection of networks like our harmonium model (or perhaps one integrated, larger network) incorporating knowledge about similarly simple circuits (e.g., a circuit with two resistors wired in parallel). These could conceivably serve as prototypes that would be invoked to deal with pieces of, or schematic versions of, larger circuits. However some powerful mechanism would still have to coordinate the parallel analyses of circuit fragments.

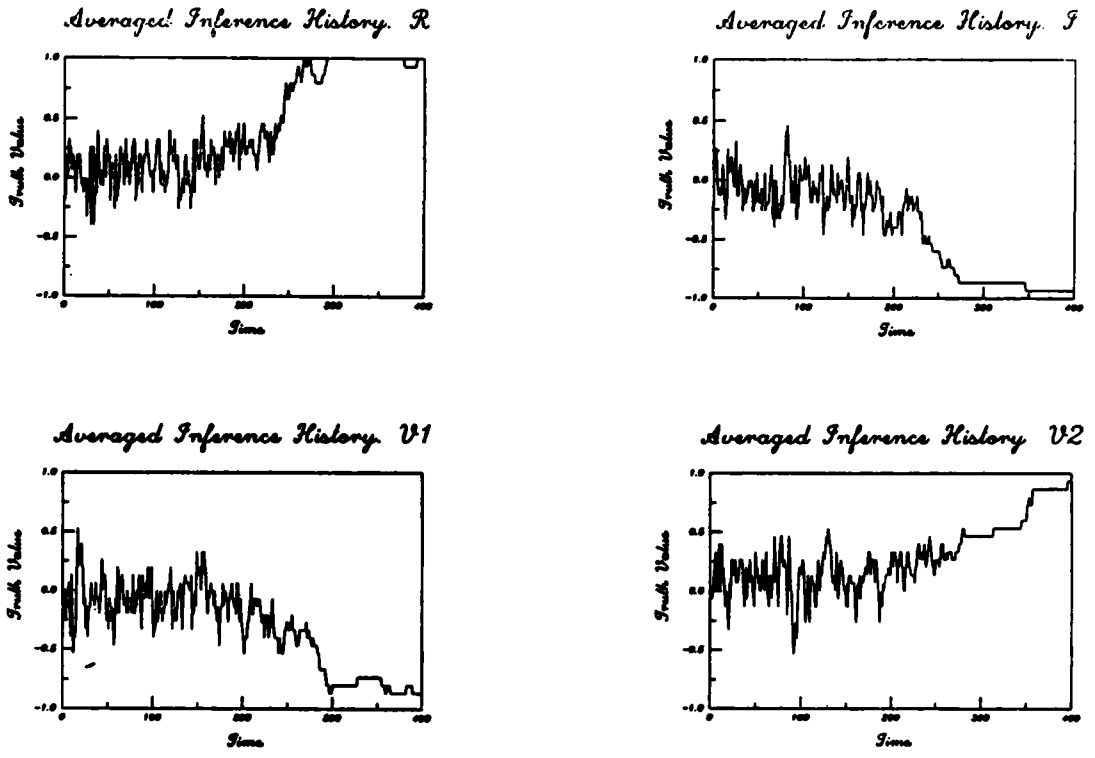



| I: | up | down | same | |
|---|---|---|---|---|
| I.c: | +1 | +1 | −1 | ["I changed"] |
| I.u: | +1 | −1 | ±1 | ["I went up"] |

Harmony function:

$$H = \sum_e A_e \sum_l \left( R_{el} B_l - \kappa |R_{el}| \right)$$

**Figure 3. Schedules for $T$ and $\kappa$, representation of *up*, *down*, *same*, and harmony function used in the harmonium model.**



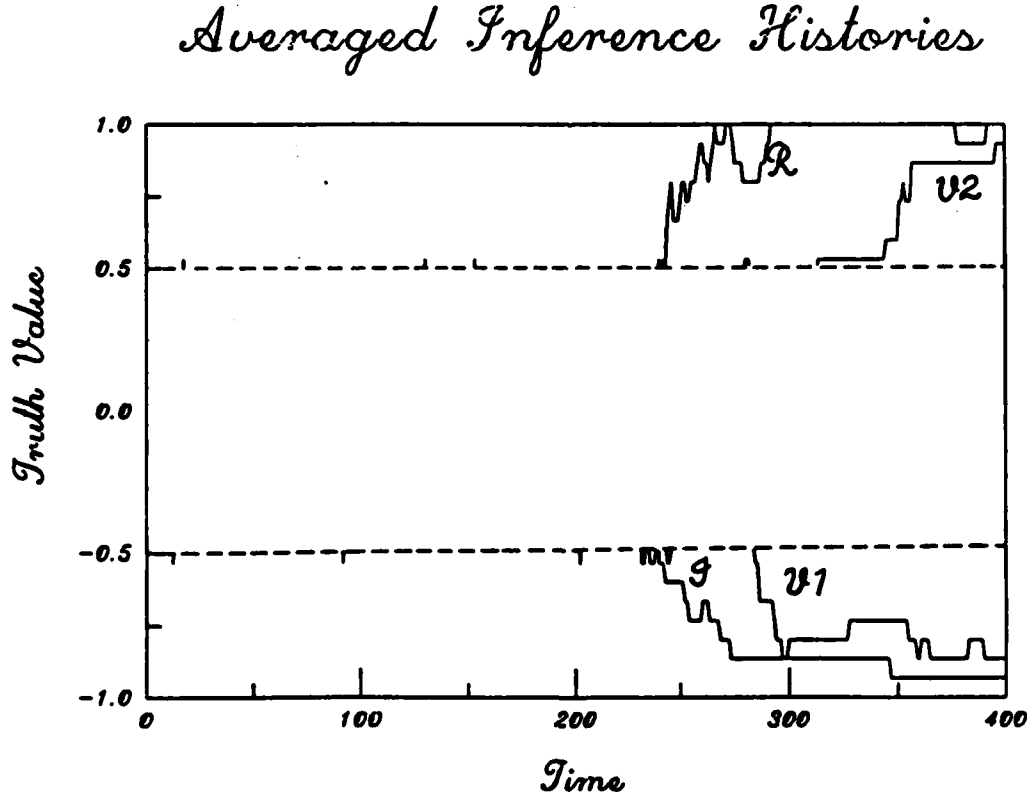




**Figure 4. Emergent sequentiality: the decisions about the direction of change of the circuit variables "freeze in" in the order $R = R_{next}$, $I = I_{next}$, $V_1$, $V_2$ ($R$ and $I$ are quite close).**

It is tempting to use a production system for this coordination, combining the strengths of the two approaches. Such a hybrid model might well be able to analyze complex circuits, but would display the production system weaknesses (lack of robustness, and so forth) in those aspects of the analysis that were relegated to the production system.

One interpretation of such a hybrid model is that the production system component is actually just a complex parallel processing network *viewed at a higher level of description*; the hybrid is of descriptive levels—there are not two independent processes. It is a major goal of ours to see if parallel models are capable of exhibiting emergent production-like behavior; the emergent seriality of the present harmonium model is an example of just such behavior.

## Discussion

The harmonium model has *implicit* knowledge of circuit laws that enable it to model naturally the nonsymbolic, intuitive component of problem solving that is difficult to model naturally with production systems and is particularly salient in expertise. At the same time it lacks the *explicit* knowledge of symbolic laws that most experts possess. Thus to model expert problem solving in general, it seems necessary to imbed the harmonium model within a hybrid parallel/production system model. We are, however, investigating whether the symbolic component of experts' processing can be preempted with conditions of very short response times, making such experimental conditions appropriate for testing the pure harmonium model. We are also planning to study unschooled electronics experts to see to what extent they are free of conscious rule application in their solution of simple circuit problems.

Much work remains to be done in analyzing the variation in the model's performance, and assessing the dependence of performance on the schedules for $T$ and $\kappa$ and the representation of the circuit. Indeed it is the development of more powerful representations within the parallel distributed processing paradigm that is the primary goal of harmony theory; by trying to enrich the knowledge of our harmonium model to incorporate more "symbol-like" explicit knowledge of circuit laws, we hope to gain more insight into how symbol manipulation might emerge from parallel distributed processing.

## References

Newell, A. (1981). Physical symbol systems. In D. A. Norman (Ed.), *Perspectives on Cognitive Science*, (pp. 37-85). Norwood, NJ: Ablex.

Riley, M. S. (in preparation). *Structural understanding in problem solving*. Unpublished doctoral dissertation. University of Pittsburgh.

Smolensky, P. (1983). Schema selection and stochastic inference in modular environments. *Proceedings of the National Conference on Artificial Intelligence*. Washington, DC.

Smolensky, P. (June, 1984). The mathematical role of self-consistency in parallel computation. To appear in the *Proceedings of the Sixth Annual Conference of the Cognitive Science Society*. Boulder, CO. Also included in this report.

# THE MATHEMATICAL ROLE OF SELF-CONSISTENCY IN PARALLEL COMPUTATION

**Paul Smolensky**

*Institute for Cognitive Science*
*University of California, San Diego*
*La Jolla, CA 92093*

**April, 1984**

**Abstract**

**By viewing the rules governing a computation as the regularities of an environment, computational states can be assigned values of self-consistency with respect to that environment. The function that measures self-consistency can play the same role in probabilistic computation as energy does in statistical physics.**

**To appear in the**
***Proceedings of the Sixth Annual Meeting of the Cognitive Science Society.***

---

**The help of David Rumelhart, Francis Crick, and other members of the UCSD Parallel Distributed Processing research group is gratefully acknowledged.**

**This research was supported by a grant from the System Development Foundation and by contract N00014-79-C-0323, NR 667-437 with the Personnel and Training Research Programs of the Office of Naval Research.**

# THE MATHEMATICAL ROLE OF SELF-CONSISTENCY IN PARALLEL COMPUTATION

## Analysis of Emergent Properties of Neural Systems

One approach to the mind/body problem is to view the description of mind as a higher level description of brain—to view psychological principles as emergent properties of neural systems. Certainly before such a view can be scientifically tested, a better understanding of both brain and mind must be established. However, enough is already known about each to make feasibility studies possible.

What methodology is capable of analyzing the emergent properties of large complex systems of interacting elements? One discipline where this analysis needs to be done is statistical physics, where large-scale properties of matter are derived mathematically from the principles believed to govern the interactions of molecular and sub-molecular constituents.

Is it possible to apply similar kinds of mathematical analysis to deduce emergent properties of neural systems? Although the principles governing neuronal interaction are by no means as well understood as those governing particles, models that abstract some of the characteristics of neural networks have been studied for some time. Hopfield (1982) has shown that with certain modifications, standard neural models can be analyzed with mathematics much like that of statistical physics, and that emergent properties can be analyzed.

One of the central concepts in statistical physics is *temperature*. The utility of this concept in performing difficult computations has been shown by Kirkpatrick et al. (1983). However the most important concept in statistical physics, as in all branches of physics, is that of *energy*. The meaning of "energy" in the computational context is not obvious; rather than a computational interpretation, Hopfield offered a general formula for the "energy" of a neural net, while Kirkpatrick et al. hand crafted "energy" formulae for their particular computations.

The application of statistical physics concepts to computation is now a rather active field of study (Hinton & Sejnowski, 1983; Hofstadter, 1983; Geman & Geman, 1983). To provide a solid foundation for this analysis, what is required in my opinion is *an interpretation of "energy" that establishes a deep connection between the formalism of statistical physics and the central problems of cognition.*

In this paper I will present the interpretation of "energy" that lies at the heart of a general computational approach I have been developing independently of the work of those interested in neural nets or in particular difficult computations. In this interpretation, *"energy" is a measure of the self-consistency of a computational state*. In place of the term "energy," which emphasizes the physical analogy, or the more technical term "Hamiltonian," which serves only to recall history and account for the physicist's notation $H$, I choose to foreground the measurement of self-consistency by using the term *harmony function*, denoted $H$. The general framework, *harmony theory*, is described in Smolensky (1984); an analysis of learning using this theory is begun in Smolensky (1983), and an application of the theory to modelling qualitative analysis of a simple electric circuit (with a discussion of the model's emergent properties) is described in Riley and Smolensky (1984). In this paper I will focus on the

computational meaning of harmony, passing quickly over other aspects of the theory. The treatment will be very informal; for more formal presentations the reader is referred to the previously cited papers.

## The Role of Harmony in Computation

Before considering how the harmony function is *defined*, we start with a discussion of how the harmony function is *used* during computation. The basic idea can be framed at a very general level. During computation, search for an answer is guided by a measure of "goodness" of possible answers: the harmony function $H$ is that measure. The search is stochastic; the computation is a Monte Carlo random walk through the solution space under the guidance of $H$. The random walk is designed so that eventually, the probability at any moment of visiting a point $p$ in the solution space is given by the *canonical distribution*:

$$\text{prob}(p) = Ne^{H(p)/T}$$

$N$ is the constant needed to normalize the probabilities so that they sum to one. $T$ is a global parameter that determines the spread in the probability distribution.

The canonical distribution is the only continuous relationship between $H$ and probability that correctly treats the independence of components of a computation. The canonical distribution also happens to be the distribution on which most of statistical physics is based. (This is no coincidence, as the notion of independent subsystem in physics maps onto that of independent subcomputations.) There is an isomorphism that maps the harmony function into minus the Hamiltonian (energy) function, and $T$ into temperature. This suggests calling $T$ the *computational temperature* of the system.

In physics, the Hamiltonian determines what states are most probable: the states with lowest energy are most probable at all temperatures, and states of high energy have negligible probability except at high temperatures. In harmony theory, the harmony function determines what states are most probable: the states with highest harmony are most probable at all computational temperatures, and states of low harmony have negligible probability except at high temperatures. $T$ can be thought of as setting the *scale* for what constitutes significant differences in harmony values. In fact, the ratio of probabilities of two states is $e^{\Delta H/T}$, where $\Delta H$ is the difference in harmony between the states. If this difference is small compared to $T$, the ratio of probabilities will be close to one; if $\Delta H$ is large compared to $T$, the state with higher harmony will be many times more probable.

The goal of the computation is to find the state of highest harmony. This means, in particular, that the state of next highest harmony should be much less likely. This requires that $T$ be small compared to the harmony difference between the two highest levels of harmony.

We could simply set $T$ to be such a low value and be done with it. However, this is not a practical search procedure. The Monte Carlo procedure will, if let run long enough, visit points with the probabilities given by the canonical distribution. However, the time required to reach this "thermal equilibrium" grows extremely rapidly as $T$ is lowered. A more practical way of zeroing in on the state of highest harmony is to start with a high temperature and gradually lower it. Early in the search, only large harmony differences are significant, and the system quickly makes a crude cut at the problem, avoiding states of extremely low harmony. As the system cools down, smaller harmony differences become significant, and more and more states are avoided as the search focuses on states with harmonies close to the maximal value. If the cooling is done gently, the state of maximal

harmony should be found in *much* less time than by giving $T$ a constant low value.

## The Relation of Harmony to the Environment

We have discussed a stochastic search technique that will find states of high harmony. But how do we design the function $H$ so that the states with high $H$ values give the correct solutions to problems? Now we must discuss the sense in which $H$ measures self-consistency.

The "correct" answer to problems are often those that satisfy a set of rules. In the circuit analysis problem considered by Riley and Smolensky (1984), for example, the rules are the physical laws of simple circuits. Any system that can correctly solve problems such as this must in some sense have a representation of the rules. In harmony theory, the rules are encoded in the harmony function. The question is, how are these rules encoded, and how can a system develop an appropriate harmony function through experience?

Of course most cognitive tasks are not as strictly governed by rules as is formal problem solving. Yet all cognition hinges on the *exploitation of regularities in the environment*, even if those regularities are less formal than Ohm's Law. Cognition enables organisms to do the *completion task*: take some limited information about the current state of their environment and make reasonable guesses about what else is likely to occur in the environment. That is, given *some* of the features that specify the environmental state, the organism can make reasonable guesses about missing features.

In harmony theory, the "rules" applied during the completion task are simply *statements that certain features can co-occur in the environment*. In the circuit application, for example, in place of a symbolic version of Ohm's Law, $V = IR$, there are many "rules" that each record a single combination of qualitative changes in $V$, $I$, and $R$ that are consistent with the law. These "rules" can in fact be thought of as *memory traces* that might be left behind by individual experiences in the environment in which the regularities hold.

Here is the general idea of how to set up a harmony function for performing the completion task in a given environment. Imagine the system experiencing many encounters with the environment; each leaves many traces that each record some of the features that co-occurred. When partial information about the current state of the environment is given in a completion problem, the harmony of a possible completion of that information is *the overall consistency between that completion and the set of all traces*. To spell this out, we consider first how the traces are determined and then how the "overall consistency" is computed.

The traces can be produced automatically by simulating exposure to an environment, or they can be produced manually by the modeller. The latter technique was used in the circuit problem: each trace was chosen to be an allowed combination of qualitative changes in the circuit quantities appearing in a single circuit law. The automatic generation of traces is yet to be explored; the idea is that traces would be produced in a random fashion (guided by the degree to which potential traces would enhance system harmony); the *statistical properties* of the resulting set of traces would then govern the emergent behavior of the system.

How is "the overall consistency between a completion and the set of all traces" computed? The idea here is that for each trace, a decision needs to be made about whether the instance it recorded is relevant to the current situation. Borrowing the usage of schema theory, a match between part of a trace and a completed set of environmental features can cause the trace to become *active*. The "overall

consistency"—the harmony—of a completion is the sum over all active traces of a measure $h$ of the degree of match between the trace and the completion. A simple definition of $h$ is the number of features in the completion that match the trace, minus the number that do not match. (A slightly more complicated definition of $h$ was used in the circuit analysis model.)

There are now two kinds of variables used in the computation: features of the environmental state, and activation values for traces. The processing has two components: computing the harmony values of possible completions, and making corresponding random decisions about which completions to visit. Computation of the harmony value requires deciding which traces to activate, and this requires computing the quality of match $h$ between traces and the completion. Just as the Monte Carlo search is used to decide what completions to visit, it can be used to decide what traces to activate. So using the traces to define the harmony of completions leads naturally to extending the search space to include both environmental feature variables *and* trace activation values.

## The Network Interpretation: A Computer Implementation

It is useful to represent the computation by a network; a portion of the network for the circuit model is shown in Figure 1. The activation variables are represented by nodes in the upper layer; each corresponds to a trace. The environmental feature variables are represented by nodes in the lower layer. There are connections between a trace variable and all the environmental features it incorporates. For simplicity, all variables (nodes) are taken to have binary values: trace activation nodes have values *active* and *inactive*; environmental feature nodes have values *present* and *absent*.

The Monte Carlo search in this network representation proceeds as follows. Initially a high temperature $T$ is chosen, all the traces are set inactive, the environmental features are permanently assigned their given values, and the remaining environmental feature variables are assigned random initial values. Then processing begins. A node is selected at random (but not one of the given features). Next the difference $\Delta H$ between the overall network harmonies that would result from the two possible values for the node is computed. In principle, this computation could be performed in the node itself, for the only quantities needed are those to which the node is connected. Finally, the node randomly selects a new value, using as the ratio of probabilities for the two values $e^{\Delta H/T}$. The process of selecting a node and selecting a value for that node is iterated while the temperature $T$ is gradually lowered according to some schedule.

The repeated selection of nodes and assignment of new values can be viewed (following Hopfield) as the asynchronous processing of processors located at the nodes and running in parallel. The relation between this parallel processing network and those considered by Hopfield and Hinton and Sejnowski is that the harmony model has a special architecture: there are two classes of nodes, and connections between but not within the two classes. The formula for harmony turns out to be minus that for Hopfield's network "energy," taking into account the special architecture and the numerical assignments *active* = 1, *inactive* = 0; *present* = 1, *absent* = −1.

## Comments on Neural Implementation

Since harmony theory is computationally-inspired, rather than neurally-inspired, the relation between the harmony network and neural networks has not been developed. However the close resemblance of the harmony network to Hopfield's neural network might suggest that harmony nodes correspond to neurons, so a brief comment is appropriate. While it does not seem unreasonable in principle to identify environmental feature nodes with neurons, it is *not* reasonable to identify trace

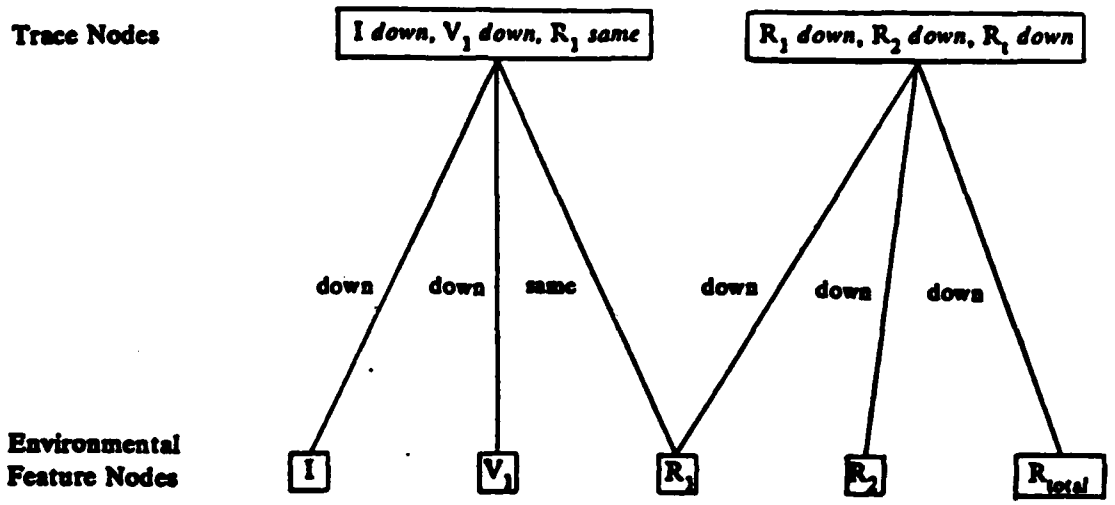



**Figure 1. A portion of the network representation of the circuit analysis model (from Riley and Smolensky, 1984). [The values *up*, *down*, *same* for environmental features (circuit variable changes) are actually represented by using two binary nodes for each variable.]**

nodes with neurons. Indeed, I imagine that each trace is distributed over the synapses of the neurons corresponding to the environmental features involved in that trace. "Activation" of the trace might correspond to a feedback-mediated rapid enhancement of the strengths of these synapses, as in von der Malsberg (1981). In this sense, even the activation of traces, a primitive operation in the theory as presently formulated, may be an emergent property of synaptic dynamics.

Even without a precise specification of the relation between harmony networks and neurons, harmony theory offers a mathematical framework within which to explore the emergence of mind from brain-like processing. The isomorphism between computation and statistical physics which it represents rests on the identification of self-consistency—harmony—as playing a central role isomorphic to that played by energy in physics.

## References

Geman, S., & Geman, D. (1983). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. Unpublished manuscript.

Hinton, G. E., & Sejnowski, T. J. (1983). Analyzing cooperative computation. *Proceedings of the Fifth Annual Conference of the Cognitive Science Society*. Rochester, NY.

Hofstadter, D. R. (1983). The architecture of Jumbo. *Proceedings of the International Machine Learning Workshop*. Monticello, IL.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences USA*, 79, 2554-8.

Kirkpatrick, S., Gelatt, C. D., Jr., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220, 671-80.

Smolensky, P. (1983). Schema selection and stochastic inference in modular environments. *Proceedings of the National Conference on Artificial Intelligence*. Washington, DC.

Smolensky, P. (1984). *Harmony theory: Thermal parallel models in a computational context*. In this report.

Riley, M. S., & Smolensky, P. (June, 1984). A parallel model of (sequential) problem solving. To appear in the *Proceedings of the Sixth Annual Conference of the Cognitive Science Society*. Boulder, CO.

von der Malsburg, C. (1981). *The correlation theory of brain function*. (Internal report 81-2). Department of Neurobiology, Göttingen, W. Germany: Max Planck Institute for Biophysical Chemistry.

# HARMONY THEORY:

# THERMAL PARALLEL MODELS IN A COMPUTATIONAL CONTEXT

**Paul Smolensky**

*Institute for Cognitive Science*
*University of California, San Diego*
*La Jolla, CA 92093*

**April, 1984**

## Abstract

**This paper discusses a particular class of parallel distributed processing models of cognition: thermal models. These models employ stochastic processors and rely on a formal mapping between parallel computation and statistical physics. A special subclass of thermal models is defined as the implementation-level description of a general mathematical framework for studying cognition: harmony theory. Harmony theory is presented at the computational and algorithmic levels as well as the implementation level.**

---

**This research was supported by a grant from the System Development Foundation and by contract N00014-79-C-0323, NR 667-437 with the Personnel and Training Research Programs of the Office of Naval Research.**

# HARMONY THEORY:

# THERMAL PARALLEL MODELS IN A COMPUTATIONAL CONTEXT

## Introduction

In recent years considerable effort has been directed at exploring computational architectures using a large number of fairly simple processors running in parallel and communicating with each other across a network of links. This style of computation, which I shall refer to as *parallel distributed processing*, has variously been called "massively parallel," "connectionistic," and "neurally-inspired." These names suggest the variety of disciplines that have found parallel distributed processing to be important to understand. Neuroscientists use the processors to model neurons; psychologists and computer scientists use parallel distributed processing to develop formal computational systems with some of the flexibility, subtlety, and power of human cognition. (For references, see Anderson & Hinton, 1981.)

The past two years have seen the emergence of a new type of parallel distributed processing that employs stochastic processors and is based on a formal mapping between parallel computation and statistical physics (Hopfield, 1982; Kirkpatrick, Gelatt, & Vecchi, 1983; Hinton & Sejnowski, 1983a, b; Hofstadter, 1983; Smolensky, 1983). I shall call cognitive models of this type *thermal models*. In this paper I present a particular class of thermal models I call *harmonium models*, with allusion to *pandemonium* (Selfridge & Neisser, 1960). I will discuss how harmonium models differ from other thermal models, as well as how thermal models differ from more traditional parallel distributed processing models, which I will refer to as *activation models*. Characteristic features of harmonium models include: an architecture that represents a process/data distinction; a global mathematical entity, the *harmony function*, that drives the processing; stochastic processors; a global system parameter, the *computational temperature*; an algorithmic process, *cooling*; and a new type of system behavior, *freezing*.

The harmony function corresponds to what others who work on thermal models have called the *energy function*, taking the term from thermal physics. [1] The harmony function, which is central to the processing of harmonium models, has an interpretation within the general context of the cognitive tasks that harmonium models are designed to perform: it measures the self-consistency of a computational state (Smolensky, 1983, 1984). This interpretation is what leads to the differences between harmonium models and other thermal models.

The central goal of this paper is to introduce a partially-developed general analytic framework I call *harmony theory*. Within this framework harmonium models emerge with a certain degree of inevitability as the description is pushed from the abstract to the implementation level. The presentation will roughly follow Marr's (1982) stratification of descriptions of computational devices into the computational, algorithmic, and implementation levels. Another level, intermediate between the computational and the algorithmic, will also be needed.

---

1. The harmony function actually corresponds to the energy function multiplied by −1.

Harmonium models of particular cognitive processes acquire additional interest when viewed within the general context of harmony theory. The models are not simply an attempt to simulate human performance, or to get a machine to "act intelligently"; they are also a vehicle for developing a general account of cognition that resides at a higher conceptual level than that of the particular models.

## Computational Level

Consider the cognitive tasks of constructing a three–dimensional percept from two-dimensional images, of constructing a coherent interpretation of a piece of text, and of solving a problem in a formal domain. These disparate tasks share an underlying structure: the filling in of missing information, using knowledge about which items fit together in the task environment. Harmony theory begins with this abstract "completion task" and a formalization of this sense of "environment," and gradually descends in abstraction to the level of implementable models of specific cognitive tasks. The ultimate goal of the research is to develop a precise characterization of a three related mathematical structures: (a) a *cognitive system*; (b) an *environment*; (c) the *completion task*. The cognitive system possesses *concepts* for representing states of the environment, and *knowledge* about which concepts fit together in the environment. The goal is to investigate: *for a given cognitive system, in a given environment, what set of concepts and knowledge relating them will enable the system to perform the completion task?*

To initiate this investigation, I formulate an appropriate formalization of the term *environment*.

Our environment can be viewed as a stream of overlapping episodes of all durations, starting at all moments of time. Cognition enables organisms to predict with some accuracy what episodes are likely to result from their actions, given the portion of those episodes about which they already have knowledge. What is critical about the environment is that *different episodes have different probabilities*. The basic cognitive task of the organism is the *prediction of likely episodes given some partial knowledge*.

For my purposes, then, I adopt these definitions. An *environment* is a probability space, the points of which are called *episodes*. In the *completion task*, information that partially specifies an episode is given as input, and the output is a set of the most probable episodes that are consistent with the input.

Many of the cognitive tasks that are studied in cognitive science can be viewed as specific instantiations of the general completion task. In the domain of story understanding, an episode is a sequence of events and actors' goals. The story partially specifies some episodes; "understanding the story" is the completion of these to full specifications, including omitted events and goals. The collection of those episodes that could possibly occur in our world, together with their corresponding probabilities, defines the story understanding environment. In the domain of visual perception, an episode is a sequence of positions of objects in three–dimensional space. Streams of two–dimensional images directly specify episodes only partially, and the job of perceptual processing is to complete those specifications.

Rather than tackle the temporal complexities of episodes, I will instead take an environment to be a probability distribution over static entities called *scenes*. The completion task then generalizes many interesting cognitive tasks that are free of time, such as understanding descriptions of static scenes and processing of single, static images.

In Riley and Smolensky (1984), harmony theory is used to study an interesting static task: qualitative analysis of the simple electric circuit shown in Figure 1. The task is to answer questions like, "What happens to the circuit if $R_2$ is increased, assuming the voltage of the battery and $R_1$ remain unchanged?" Here a "scene" is a set of *qualitative changes in circuit features*. Those sets of changes that are consistent with the circuit laws of elementary physics define the scenes that have nonzero probability in this environment. The given question specifies *some* of the qualitative changes defining a scene, namely, the changes in the resistances and in the battery's voltage. The task is to complete this to a full specification of a "highly probable" scene, i.e., fill in the appropriate changes in the other circuit quantities like currents and voltage drops. If the information given in the problem uniquely determines a "correct answer," then, given the input, one scene has probability one and the others have probability zero.

The probability distribution for the environment of elementary physics problems is artificial; there is sharp distinction between scenes that are "allowed" and those that are not, i.e., between those that have nonzero probability and those that have zero probability. This characterizes a *formal* environment, one that can be delimited strictly by formal rules like the circuit laws. While conventional computers are at home in such environments, people are not; at least one can argue that more training is required for people to perform well in formal environments than in informal ones. A central empirical question for this approach is: *What are the properties of the environments in which natural cognitive systems can actually perform the completion task with some accuracy?* At this stage, intuition must serve in place of an empirical answer. The environments are most likely *sparse*, with a huge fraction of the space of all possible scenes having an extremely low probability. Furthermore, we perceive scenes (in my generalized sense of the word) as groups of entities which are in turn groups of sub–entitites, and so on. This suggests that the environments with which the human mind is designed to deal exhibit a kind of *modularity*: the probability of a scene can be computed by describing it in terms of modules of various scales and recursively computing the probability that the modules at one scale would be combined to make the modules of the next larger scale.

Concepts, according to this intuition, correspond to the modules in the environment. Knowledge about these concepts is what enables us to compute the probabilities of various combinations of concepts.

To formally define, *at this computational level*, modular environments, cognitive systems, concepts, and knowledge is a major goal of this research. At the moment, however, precise characterizations corresponding to these notions exist only at the algorithmic level. The next section describes the intuitions that lie between the algorithmic–level description and a yet-to-be formulated computational–level description.

## Computational/Algorithmic Level

A completion task can be performed, it is assumed, because prior experience with the environment has left traces of statistical connections between the information that is given and the information that must be filled in. The mechanisms which might maintain such traces in the brain are

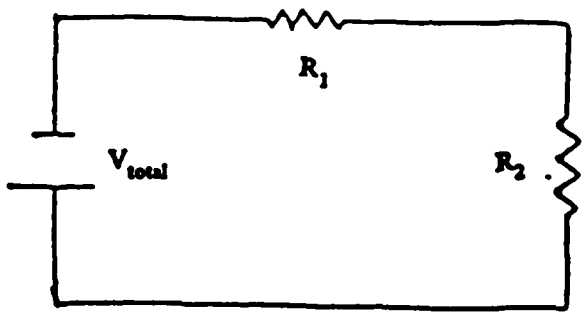



Figure 1. A series circuit with two resistors, $R_1$ and $R_2$. What are the effects of an increase in the resistance of $R_2$, assuming that $V_{total}$ and the resistance of $R_1$ remain the same?

*not* considered. [2] Rather, I shall give a strictly formal description of a cognitive system that possesses such traces. This description presupposes a representation of environmental scenes within the cognitive system.

Each scene in the environment is assumed to be described in the cognitive system by a set of *representation variables* $\{R_1, R_2, \cdots, R_N\}$. Each $R_i$ is taken (for simplicity) to have value *true* or *false* for each scene. In the circuit analysis problem, for example, $R_1$ and $R_2$ might represent whether there is a change in the battery's voltage and current, respectively; each of these will be either true or false for each scene. The representational variables as a whole are assumed to support representations of scenes at all the levels of abstraction employed by the cognitive system; assigning values to some of the variables may require considerable processing.

The list of true/false $R_i$ values for a scene will be called its *representation vector* $\mathbf{R}$. Each $\mathbf{R}$ has some probability in a given environment. In the completion task, some of the elements of an $\mathbf{R}$ vector are specified as input (e.g., $R_1$), and as output the system must give values for the remaining elements (e.g., $R_2$).

The crucial question for solving the completion task is, what values of various $R_i$ variables go together in the environment? A cognitive system must accumulate the knowledge that answers this question as it experiences a sample of vectors $\mathbf{R}$. Each $\mathbf{R}$ is assumed to leave many traces, which are simply copies of *pieces* of $\mathbf{R}$. Each trace records a single co-ocurrence of the specific values of the $R_i$ variables present in that trace. After considerable experience with the environment, an ensemble of traces is built up; this ensemble implicitly encodes the environment's probability distribution.

In this paper we shall not analyze the important question of which pieces of $\mathbf{R}$ are maintained as traces; the issue is considered in Smolensky (1983). Very roughly, the idea is that each trace records one of the *modules* present in $\mathbf{R}$. [3] For present purposes we assume that the set of traces encoding the system's knowledge of the environment has been produced either by an unspecified training process or by explicit design of the modeller. In the electric circuit problem, for example, the traces are put in by hand; each one consists of one of the possible instances of changes of circuit quantities that are consistent with one of the circuit laws. (For instance, one of the traces records the co-occurrence of: (a) a decrease in the current, (b) a decrease in the voltage drop across resistor 1, and (c) no change in the resistance of resistor 1; this is one instance of Ohm's Law for resistor 1.)

The traces are encoded as a set of *trace vectors* $\{\mathbf{T}_1, \mathbf{T}_2, \cdots, \mathbf{T}_M\}$. Each $\mathbf{T}_a$ is a piece of some representation vector $\mathbf{R}$, i.e., a set of true/false values for a *subset* of the representation variables. $\mathbf{T}_a$ is viewed as a vector of values, one for *each* $R_i$: either *true*, *false*, or *unspecified*.

---

2. A few remarks on possible neural implementations of harmony theory may be found in Smolensky (1984).

3. A little more precisely, the idea is this. For each $\mathbf{R}$ experienced by the system, some mechanism transcribes pieces of $\mathbf{R}$ and records them as traces. After experiencing a large number of scenes $\mathbf{R}$, a large collection of traces will have accumulated. Some traces will be duplicated many times, others less often. Those frequently duplicated define the primary *concepts* in terms of which scenes will be processed by the system. Thus the system's concepts *emerge from the statistical properties* of the traces. Investigation of appropriate mechanisms for recording traces must therefore center on analyzing the statistical properties of the resulting traces in various environments. What criteria determine whether the traces produced by some recording mechanism have reasonable statistics? The primary criterion is the system's performance on the completion task enabled by those traces. A simpler criterion thought to underly good performance involves the concept of *harmony* to be discussed below; this *training harmony* criterion is considered in Smolensky (1983).

Loosely, here is how completions are performed using the traces $\mathbf{T}_a$. The system checks to see which traces are consistent with the input; these become *active*. When active traces specify true/false values for missing information, these values are used to decide how to fill in the missing information.

Thus, to each trace vector $\mathbf{T}_a$ is associated an *activation value* $A_a$. The list of all activation values forms the *activation vector* **A**. For simplicity each $A_a$ will be taken to be either *active* or *inactive*; these binary values will be sufficient for our purposes (unlike with traditional activation models).

Actually, the assignment of true/false values to missing information is done in parallel with the assignment of active/inactive values to the trace variables. To carry out the completion, the system simultaneously performs searches in the space of all vectors **A** and in the space of all vectors **R** that contain the true/false values given by the input.

The goal of the search is to find those completions **R** that are highly probable in the environment. Intuitively, there is a relationship between these **R** and the traces $\mathbf{T}_a$. Highly probable completions are those containing combinations of modules that occur frequently in the environment. Such a completion will be highly consistent with many traces. This suggests using, in lieu of a literal computation of the probability of a completion **R**, a measure of its *goodness* that counts the number of traces with which **R** is consistent, and the overall degree of that consistency.

A convenient representation for this measure of goodness incorporates the idea that traces that are consistent with the input should be active. For any activation vector **A** and completion **R**, define[4]

$$H(\mathbf{A},\mathbf{R}) = \sum_a A_a \, \mathbf{R} \cdot \mathbf{T}_a$$

Here the following numerical assignments are used: *true* = 1, *false* = −1, *unspecified* = 0; *active* = 1, *inactive* = 0. $\cdot$ is the inner product: $\mathbf{R} \cdot \mathbf{T}_a = \sum_i R_i (T_a)_i$; this is just the number of representational variables $R_i$ whose values agree with the corresponding values in $\mathbf{T}_a$, minus the number that disagree.

$H(\mathbf{A},\mathbf{R})$ measures how consistent **R** is with the traces active in **A**: $H$ is called the *harmony function*. $H$ is the central player in harmony theory because all the decision-making in the system is driven towards achieving maximal consistency, i.e., harmony. If a trace is consistent with the input, it becomes active because doing so raises the harmony. If setting a representational variable to a true or false value is consistent with active traces, that assignment is made because it raises the harmony. In short, the space of activation vectors **A** and completions **R** is searched to find the values that achieve high harmony. These should be the completions that are most probable in the environment.

An important goal of the theory is to provide a mathematical characterization of modular environments that allows a proof that high harmony completions are high probability completions. At the moment, this identification rests on intuition.

In a sense, the traces serve during search as *anti-constraints*. The good solutions are those that

---

4. In Smolensky (1983) and Riley and Smolensky (1984), a somewhat more complex formula for $H$ is used. Axioms defining the properties a harmony function must satisfy will be an important part of the formulation of harmony theory at the computational level.

*satisfy* a number of anti-constraints, i.e., match a number of traces. This has many similarities to constraint-based search, in which good solutions are those that *fail to violate* a number of constraints. However, the differences are more than technical. It is easy to see how a system can *learn* anti-constraints: these are simply the traces left by experience, records that show what variable values *can* go together. It is not so easy to learn what values *cannot* go together, when a "teacher" or reinforcer is absent. Furthermore, if environmental probability distributions are sparse, as discussed above, then it seems more feasible to record *allowed* co-occurrences than *forbidden* ones.

## Algorithmic Level

Having characterized good completions at the computational level (high probability) and at the computational/algorithmic level (high harmony), it is time to specify an algorithm that will construct good completions. The algorithm used in harmony theory is a stochastic one: it constructs a completion with a probability related to its harmony. The only mathematically viable relationship (Smolensky, 1983) is the *canonical distribution*:

$$\text{prob}(\mathbf{A}, \mathbf{R}) = n\, e^{H(\mathbf{A},\mathbf{R})/T}$$

Here $n$ is a normalization constant, and $T$ is a positive system parameter. If two completions have different harmonies, the more harmonious one will be more probable; if the harmony difference $\Delta H$ is large compared to $T$, then the ratio of probabilities ($e^{\Delta H/T}$) will be large. Thus the greater $T$, the less will be the bias in favor of the most harmonious completions, and the more random the completions will seem. The randomness parameter $T$ is called the *computational temperature* because its role in the canonical distribution is identical to that of physical temperature in the strictly isomorphic canonical distribution (Boltzmann law) of statistical mechanics.

Good completions will be overwhelmingly likely if and only if the temperature is very low. Thus to achieve good performance a low temperature is needed.

In statistical mechanics there is a well-known Monte Carlo search method (the "heat bath algorithm") that can be used to stochastically explore the problem space of vectors A and R, visiting points in the space with the probabilities of the canonical distribution (Metropolis, Rosenbluth, Rosenbluth, Teller, & Teller, 1953; Binder, 1979). This algorithm starts at a random point, and randomly chooses a possible direction of travel. The change in harmony $\Delta H$ that would result from a step in that direction is computed; the decision to take the step is then made randomly, with likelihood ratio for taking or not taking the step set equal to $e^{\Delta H/T}$. "Choosing a possible direction of travel" amounts to selecting a *single* variable $A_a$ or $R_i$; "taking a step" amounts to changing the binary value of the selected variable.

The process of choosing a direction and deciding whether to take a step is iterated. It can be proved that, eventually, the probability of being at a point is given by the canonical distribution. The higher $T$, the more quickly this "thermal equilibrium" is reached.

The practical difficulty with this algorithm is that for the low $T$ values needed to get good completions, it takes an unacceptably long time to reach thermal equilibrium. A way to get to good completions faster is to start with a fairly high temperature, and *cool the system down* during the computation (Kirkpatrick, Gelatt, & Vecchi, 1983). Cooling (or "simulated annealing") is a new computational process characteristic of thermal models; it brings with it a new computational behavior: *freezing*. As the temperature is lowered, various system variables lock in to values which

become very resistant to change.

What is happening during the cooling process is that $T$, the scale on which the significance of harmony differences is measured, is getting smaller and smaller. Early in the processing, only big harmony differences matter: the system avoids only states of very low relative harmony; a very crude cut is made at the problem. A large region of the problem space is explored. As the processing continues, the system gets more exacting, narrowing the search to states with harmony values that are closer and closer to the maximum attainable value. This kind of search has the advantage that whenever its results are examined, their reliability is as high as can be achieved in the elapsed time (loosely speaking).

A good theoretical understanding of how to regulate the cooling will be difficult to achieve. At the moment, certain techniques allow estimates of freezing temperatures in simple situations, but cooling schedules are still largely defined in ad hoc ways.

## Implementation Level

The most natural implementation for harmony theory uses a parallel computing network. This parallel device, *harmonium*, is easily simulated on a serial computer.

To implement the Monte Carlo search algorithm discussed above, we set up one processor for each of the variables $A_\alpha$ and $R_i$. According to the discussion in the previous section, the values of the activation processors are 1 and 0, while the values for the representation processors are +1 and −1. This set of values is not typical of thermal models.

The algorithm first involves randomly picking a direction in search space; this amounts to picking one of the processors. To ensure that only one processor changes its value at a time, the processors are assumed to take a random amount of time to make their decisions and then instantaneously make their change; the probability of simultaneous changes is then zero. Once a change is made, the new value must be available immediately. This type of asynchronous updating is not typical of activation models, but is typical of thermal models (Hopfield, 1982).

The algorithm requires that to decide on its new value, a processor must compute the harmony change $\Delta H$ that would result from changing its value. To perform this computation, a given processor must be connected to others in order to read their values. The required pattern of interconnections, found by inspection of the harmony function, is graphically summarized in Figure 2, in which each node is a processor. $\Delta H$ for a given node is a weighted sum of the values of the nodes connected to it; the weight linking representation node $i$ and trace node $\alpha$ is $(\mathbf{T}_\alpha)_i$, the $i^{\text{th}}$ element of the vector $\mathbf{T}_\alpha$. This weight applies to values passed in *either* direction; bidirectional weights are characteristic of thermal models.

The architecture shown in Figure 2 is not typical of thermal or activation models. The purpose of the processing is to set up the appropriate completion on the *representation nodes*; the trace nodes serve solely to mediate between represention nodes, which are not directly interconnected. It is useful to regard the representation nodes as a data blackboard, and the trace vectors $\mathbf{T}_\alpha$ (or equivalently the pattern of connections) as the program, and the activation values as internal program variables.

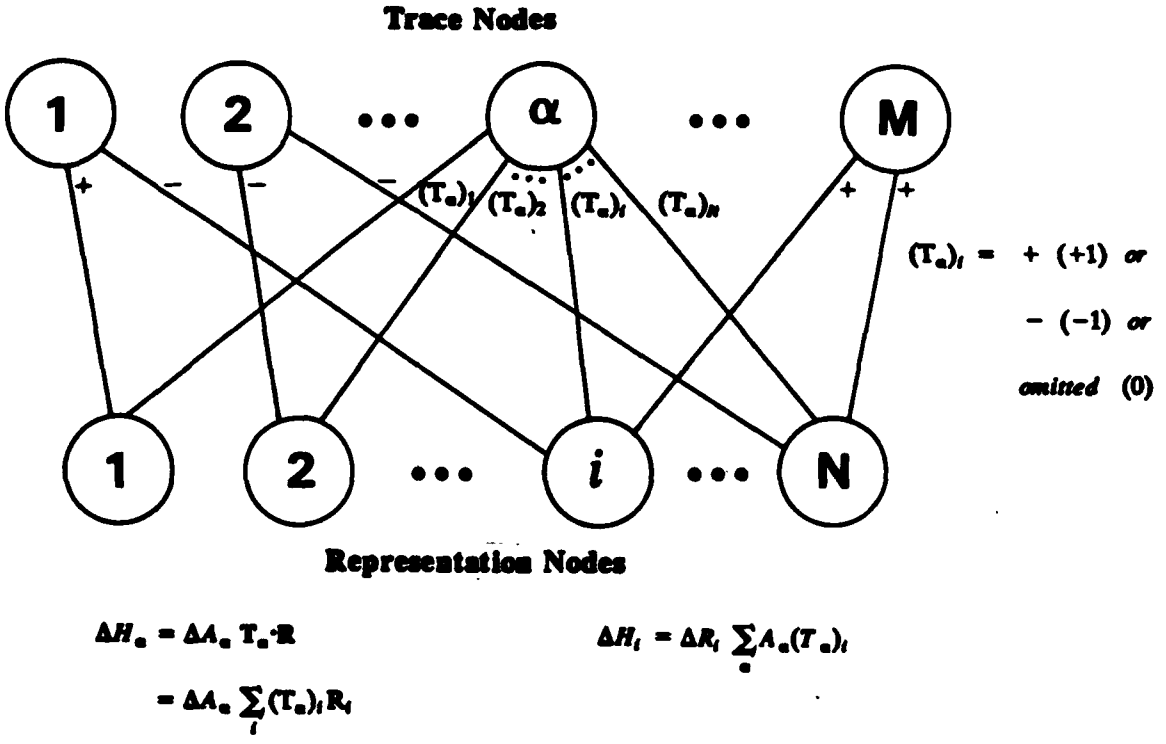
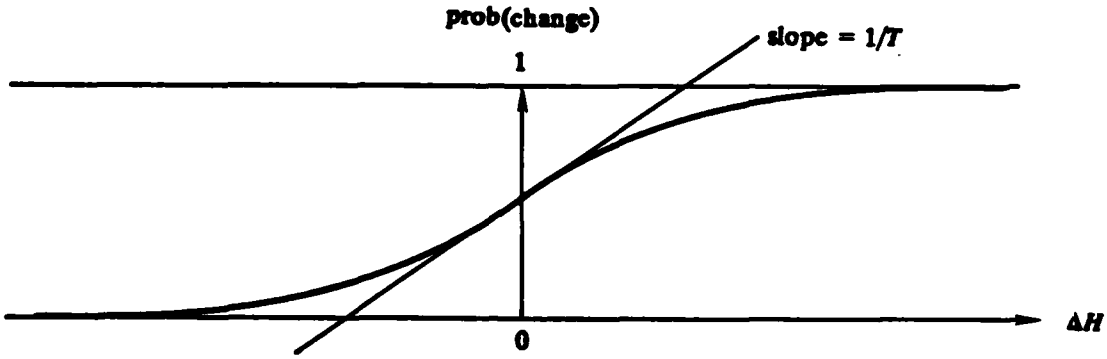



**Figure 2. A graphical representation of harmonium. The nodes denote stochastic processors, and the links denote communication lines.**

In particular it should be noted that the traditional hierarchical picture of successive vertical layers of nodes with layer-to-layer vertical connections does not exist here. One can visualize the representational nodes being laid out with abstract representational variables to the right and concrete variables to the left. Input at the leftmost nodes activates traces that are connected to them; these traces are also connected to slightly more abstract representational nodes, which are then assigned values consistent with the active traces. This in turn activates new traces, and so it goes. Decision-making passes rightward, bouncing back and forth between the lower to upper layers. In place of a pre-wired, rigid vertical hierarchical architecture is a fluid horizontal architecture that can implement a hierarchy when appropriate. (In fact, the statistical properties of the participation of "nodes on the right" in the traces that accumulate during experience with an environment are what determine the "abstract concepts" that dynamically evolve in that environment.)

Once a node has computed the difference in harmony $\Delta H$ between its two possible states, the likelihood ratio for adopting its two states is $e^{\Delta H/T}$. Converting this to the absolute probability of changing value gives the result shown in Figure 3. This sort of sigmoidal relation between the weighted sum of the inputs to a node and its decision is common in activation models; however two differences should be noted. First, in activation models the ordinate of Figure 3 would be a continuous node value; here, it is a *probability* for a discrete node value. Second, the slope of the sigmoidal curve at the origin, $1/T$, is not fixed; it increases as the computation proceeds.

The processing features visible at the implementation level—i.e., the defining properties of harmonium—are all strict consequences of the algorithmic-level analysis of harmony theory. Ongoing development of the theory is aimed at filling the logical and empirical gaps linking the analysis at the algorithmic level to those at higher levels, and gaining experience applying harmonium to specific cognitive tasks.




$$\text{prob(change)} = \frac{e^{\Delta H/T}}{1 + e^{\Delta H/T}}$$

**Figure 3. The sigmoidal relation between the weighted sum of inputs to a harmonium node ($\Delta H$) and the probability that the node changes its value.**

## References


Anderson, J. A., & Hinton, G. E. (1981). Models of information processing in the brain. In G. E. Hinton and J. A. Anderson, *Parallel models of associative memory*. Hillsdale, NJ: Erlbaum.

Binder, K. (ed.) (1979). *Monte Carlo methods in statistical physics*. Berlin: Springer-Verlag.

Geman, S., & Geman, (1983). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. Unpublished manuscript.

Hinton, G. E., & Sejnowski, T. J. (1983a). Analyzing cooperative computation. *Proceedings of the Fifth Annual Conference of the Cognitive Science Society*. Rochester, NY.

Hinton, G. E., & Sejnowski, T. J. (1983b). Optimal perceptual inference. *Proceedings of the I.E.E.E. Conference on Computer Vision and Pattern Recognition*. Washington, DC.

Hofstadter, D. R. (1983). The architecture of Jumbo. *Proceedings of the International Machine Learning Workshop*. Monticello, IL.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences USA, 79*, 2554-8.

Kirkpatrick, S., Gelatt, C. D., Jr., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science, 220*, 671-80.

Marr, D. (1982). *Vision*. San Francisco: Freeman.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). *J. Chem. Phys., 21*, 1087.

Riley, M. S. & Smolensky, P. (1984). A parallel model of (sequential) problem solving. To appear in the *Proceedings of the Sixth Annual Conference of the Cognitive Science Society*. Boulder, CO. Also included in this report.

Selfridge, O. G., & Neisser, U. (1960). Pattern recognition by machine. *Scientific American, 203*, 60-68.

Smolensky, P. (1983). Schema selection and stochastic inference in modular environments. *Proceedings of the National Conference on Artificial Intelligence*. Washington, DC.

Smolensky, P. (1984). The mathematical role of self-consistency in parallel computation. To appear in the *Proceddings of the Sixth Annual Conference of the Cognitive Science Society*. Boulder, CO. Also included in this report.

## Cognitive Science ONR Technical Report List

The following is a list of publications by people in the Cognitive Science Lab and the Institute for Cognitive Science. For reprints, write or call:

Institute for Cognitive Science, C-015
University of California, San Diego
La Jolla, CA 92093
(619) 452-6771

8001. Donald R. Gentner, Jonathan Grudin, and Eileen Conway. *Finger Movements in Transcription Typing*. May 1980.

8002. James L. McClelland and David E. Rumelhart. *An Interactive Activation Model of the Effect of Context in Perception: Part I*. May 1980.

8003. David E. Rumelhart and James L. McClelland. *An Interactive Activation Model of the Effect of Context in Perception: Part II*. July 1980.

8004. Donald A. Norman. *Errors in Human Performance*. August 1980.

8005. David E. Rumelhart and Donald A. Norman. *Analogical Processes in Learning*. September 1980.

8006. Donald A. Norman and Tim Shallice. *Attention to Action: Willed and Automatic Control of Behavior*. December 1980.

8101. David E. Rumelhart. *Understanding Understanding*. January 1981.

8102. David E. Rumelhart and Donald A. Norman. *Simulating a Skilled Typist: A Study of Skilled Cognitive-Motor Performance*. May 1981.

8103. Donald R. Gentner. *Skilled Finger Movements in Typing*. July 1981.

8104. Michael I. Jordan. *The Timing of Endpoints in Movement*. November 1981.

8105. Gary Perlman. *Two Papers in Cognitive Engineering: The Design of an Interface to a Programming System and MENUNIX: A Menu-Based Interface to UNIX (User Manual)*. November 1981.

8106. Donald A. Norman and Diane Fisher. *Why Alphabetic Keyboards Are Not Easy to Use: Keyboard Layout Doesn't Much Matter*. November 1981.

8107. Donald R. Gentner. *Evidence Against a Central Control Model of Timing in Typing.* December 1981.

8201. Jonathan T. Grudin and Serge Larochelle. *Digraph Frequency Effects in Skilled Typing.* February 1982.

8202. Jonathan T. Grudin. *Central Control of Timing in Skilled Typing.* February 1982.

8203. Amy Geoffroy and Donald A. Norman. *Ease of Tapping the Fingers in a Sequence Depends on the Mental Encoding.* March 1982.

8204. LNR Research Group. *Studies of Typing from the LNR Research Group: The role of context, differences in skill level, errors, hand movements, and a computer simulation.* May 1982.

8205. Donald A. Norman. *Five Papers on Human-Machine Interaction.* May 1982.

8206. Naomi Miyake. *Constructive Interaction.* June 1982.

8207. Donald R. Gentner. *The Development of Typewriting Skill.* September 1982.

8208. Gary Perlman. *Natural Artificial Languages: Low-Level Processes.* December 1982.

8301. Michael C. Mozer. *Letter Migration in Word Perception.* April 1983.

8302. David E. Rumelhart and Donald A. Norman. *Representation in Memory.* June 1983.

8303. The HMI Project at University of California, San Diego. *User Centered System Design: Part I, Papers for the CHI 1983 Conference on Human Factors in Computer Systems.* November 1983.

8304. Paul Smolensky. *Harmony Theory: A Mathematical Framework for Stochastic Parallel Processing.* December 1983.

8401. Stephen W. Draper and Donald A. Norman. *Software Engineering for User Interfaces.* January 1984.

8402. The UCSD HMI Project. *User Centered System Design: Part II.* March 1984.

8403. Paul Smolensky and Mary S. Riley. *Harmony Theory: Problem Solving, Parallel Cognitive Models, and Thermal Physics.* April 1984.

## ICS Technical Report List

**The following is a list of publications by people in the Institute for Cognitive Science. For reprints, write or call:**

**Institute for Cognitive Science, C-015**
**University of California, San Diego**
**La Jolla, CA 92093**
**(619) 452-6771**

8301. David Zipser. ***The Representation of Location.*** May 1983.

8302. Jeffrey Elman & Jay McClelland. ***Speech Perception as a Cognitive Process: The Interactive Activation Model.*** April 1983.

8303. Ron Williams. ***Unit Activation Rules for Cognitive Networks.*** November 1983.

8304. David Zipser. ***The Representation of Maps.*** November 1983.

8305. The HMI Project. ***User Centered System Design: Part I, Papers for the CHI '83 Conference on Human Factors in Computer Systems.*** November 1983.

8306. Paul Smolensky. ***Harmony Theory: A Mathematical Framework for Stochastic Parallel Processing.*** December 1983.

8401. Stephen W. Draper and Donald A. Norman. ***Software Engineering for User Interfaces.*** January 1984.

8402. The UCSD HMI Project. ***User Centered System Design: Part II.*** March 1984.

8403. Steven L. Greenspan. ***Reference Comprehension: A Topic-Comment Analysis of Sentence-Picture Verification.*** April 1984.

8404. Paul Smolensky and Mary S. Riley. ***Harmony Theory: Problem Solving, Parallel Cognitive Models, and Thermal Physics.*** April 1984.

Navy

1 Robert Ahlers
Code N711
Human Factors Laboratory
NAVTRAEQUIPCEN
Orlando, FL 32813

1 Dr. Ed Aiken
Navy Personnel R&D Center
San Diego, CA 92152

1 Dr. Robert Blanchard
Navy Personnel R&D Center
San Diego, CA 92152

1 Dr. Nick Bond
Office of Naval Research
Liaison Office, Far East
APO San Francisco, CA 96503

1 Dr. Richard Cantone
Navy Research Laboratory
Code 7510
Washington, DC 20375

1 Dr. Fred Chang
Navy Personnel R&D Center
San Diego, CA 92152

1 Dr. Stanley Collyer
Office of Naval Technology
800 N. Quincy Street
Arlington, VA 22217

1 CDR Mike Curran
Office of Naval Research
800 N. Quincy St.
Code 270
Arlington, VA 22217

1 Dr. Jude Franklin
Code 7510
Navy Research Laboratory
Washington, DC 20375

1 Dr. Jim Hollan
Code 14
Navy Personnel R & D Center
San Diego, CA 92152

1 Dr. Ed Hutchins
Navy Personnel R&D Center
San Diego, CA 92152



Navy

1 Dr. Norman J. Kerr
Chief of Naval Technical Training
Naval Air Station Memphis (75)
Millington, TN 38054

1 Dr. Peter Kincaid
Training Analysis & Evaluation Group
Dept. of the Navy
Orlando, FL 32813

1 Dr. William L. Maloy (02)
Chief of Naval Education and Training
Naval Air Station
Pensacola, FL 32508

1 Dr. Joe McLachlan
Navy Personnel R&D Center
San Diego, CA 92152

1 Dr. James McMichael
Navy PersonnelR&D Center
San Diego, CA 92152

1 Dr William Montague
NPRDC Code 13
San Diego, CA 92152

1 Library, Code P201L
Navy Personnel R&D Center
San Diego, CA 92152

1 Technical Director
Navy Personnel R&D Center
San Diego, CA 92152

6 Commanding Officer
Naval Research Laboratory
Code 2627
Washington, DC 20390

1 Office of Naval Research
Code 433
800 N. Quincy SStreet
Arlington, VA 22217

6 Personnel & Training Research Group
Code 442PT
Office of Naval Research
Arlington, VA 22217

Navy

1 Dr. Gary Poock
Operations Research Department
Code 55PK
Naval Postgraduate School
Monterey, CA 93940

1 Dr. Robert G. Smith
Office of Chief of Naval Operations
OP-987H
Washington, DC 20350

1 Dr. Alfred F. Smode, Director
Department N-7
Naval Training Equipment Center
Orlando, FL 32813

1 Dr. Richard Snow
Liaison Scientist
Office of Naval Research
Branch Office, London
Box 39
FPO New York, NY 09510

1 Dr. Richard Sorenson
Navy Personnel R&D Center
San Diego, CA 92152

1 Dr. Frederick Steinheiser
CNO - OP115
Navy Annex
Arlington, VA 20370

1 Dr. Thomas Sticht
Navy Personnel R&D Center
San Diego, CA 92152

1 Roger Weissinger-Baylon
Department of Administrative Sciences
Naval Postgraduate School
Monterey, CA 93940

1 Dr John H. Wolfe
Navy Personnel R&D Center
San Diego, CA 92152

1 Dr. Wallace Wulfeck, III
Navy Personnel R&D Center
San Diego, CA 92152

Marine Corps

1 H. William Greenup
Education Advisor (E031)
Education Center, MCDEC
Quantico, VA 22134

1 Special Assistant for Marine
Corps Matters
Code 100M
Office of Naval Research
800 N. Quincy St.
Arlington, VA 22217

1 DR. A.L. SLAFKOSKY
SCIENTIFIC ADVISOR (CODE RD-1)
HQ, U.S. MARINE CORPS
WASHINGTON, DC 20380

ONR Distribution List

Army

1 Technical Director
U. S. Army Research Institute for the Behavioral and Social Sciences
5001 Eisenhower Avenue
Alexandria, VA 22333

1 Dr. Beatrice J. Farr
U. S. Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333

1 Dr. Harold F. O'Neil, Jr.
Director, Training Research Lab
Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333

1 Commander, U.S. Army Research Institute for the Behavioral & Social Sciences
ATTN: PERI-BR (Dr. Judith Orasanu)
5001 Eisenhower Avenue
Alexandria, VA 22333

1 Joseph Psotka, Ph.D.
ATTN: PERI-IC
Army Research Institute
5001 Eisenhower Ave.
Alexandria, VA 22333

1 Dr. Robert Sasmor
U. S. Army Research Institute for the Behavioral and Social Sciences
5001 Eisenhower Avenue
Alexandria, VA 22333

Air Force

1 U.S. Air Force Office of Scientific Research
Life Sciences Directorate, NL
Bolling Air Force Base
Washington, DC 20332

1 Bryan Dallman
AFHRL/LRT
Lowry AFB, CO 80230

1 Dr. Alfred R. Fregly
AFOSR/NL
Bolling AFB, DC 20332

1 Dr. Genevieve Haddad
Program Manager
Life Sciences Directorate
AFOSR
Bolling AFB, DC 20332

1 Dr. John Tangney
AFOSR/NL
Bolling AFB, DC 20332

1 Dr. Joseph Yasutake
AFHRL/LRT
Lowry AFB, CO 80230



Department of Defense

12 Defense Technical Information Center
Cameron Station, Bldg 5
Alexandria, VA 22314
Attn: TC

1 Military Assistant for Training and Personnel Technology
Office of the Under Secretary of Defens for Research & Engineering
Room 3B129, The Pentagon
Washington, DC 20301

1 Major Jack Thorpe
DARPA
1400 Wilson Blvd.
Arlington, VA 22209

1 Dr. Robert A. Wisher
OUSDRE (ELS)
The Pentagon, Room 3B129
Washington, DC 20301

Civilian Agencies

1 Dr. Patricia A. Butler
NIE-BRN Bldg, Stop 8 7
1200 19th St., NW
Washington, DC 20208

1 Dr. Andrew R. Molnar
Office of Scientific and Engineering Personnel and Education
National Science Foundation
Washington, DC 20550

1 Dr. Everett Palmer
Mail Stop 239-3
NASA-Ames Research Center
Moffett Field, CA 94035

1 Dr. Mary Staddard
C 10, Mail Stop B296
Los Alamos National Laboratories
Los Alamos, NM 87545

1 Dr. Edward C. Weiss
National Science Foundation
1800 G Street, NW
Washington, DC 20550

1 Dr. Joseph L. Young, Director
Memory & Cognitive Processes
National Science Foundation
Washington, DC 20550

Private Sector

1 Dr. John R. Anderson
Department of Psychology
Carnegie-Mellon University
Pittsburgh, PA 15213

1 Dr. Alan Baddeley
Medical Research Council
Applied Psychology Unit
15 Chaucer Road
Cambridge CB2 2EF
ENGLAND

1 Eva L. Baker
Director
UCLA Center for the Study of Evaluation
145 Moore Hall
University of California, Los Angeles
Los Angeles, CA 90024

1 Mr. Avron Barr
Department of Computer Science
Stanford University
Stanford, CA 94305

1 Dr. Menucha Birenbaum
School of Education
Tel Aviv University
Tel Aviv, Ramat Aviv 69978
Israel

1 Dr. John Black
Yale University
Box 11A, Yale Station
New Haven, CT 06520

1 Dr. John S. Brown
XEROX Palo Alto Research Center
3333 Coyote Road
Palo Alto, CA 94304

1 Dr. Glenn Bryan
6208 Poe Road
Bethesda, MD 20817

1 Dr. Bruce Buchanan
Department of Computer Science
Stanford University
Stanford, CA 94305

1 Dr. Jaime Carbonell
Carnegie-Mellon University
Department of Psychology
Pittsburgh, PA 15213



Private Sector

1 Dr. Pat Carpenter
Department of Psychology
Carnegie-Mellon University
Pittsburgh, PA 15213

1 Dr. Davida Charney
Department of Psychology
Carnegie-Mellon University
Schenley Park
Pittsburgh, PA 15213

1 Dr. Micheline Chi
Learning R & D Center
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 15213

1 Dr. William Clancey
Department of Computer Science
Stanford University
Stanford, CA 94306

1 Dr. Michael Cole
University of California
at San Diego
Laboratory of Comparative
Human Cognition - D003A
La Jolla, CA 92093

1 Dr. Allan M. Collins
Bolt Beranek & Newman, Inc.
50 Moulton Street
Cambridge, MA 02138

1 Dr. Emmanuel Donchin
Department of Psychology
University of Illinois
Champaign, IL 61820

1 Dr. Thomas H. Duffy
Department of English
Carnegie-Mellon University
Schenley Park
Pittsburgh, CA 15213

1 ERIC Facility-Acquisitions
4833 Rugby Avenue
Bethesda, MD 20014

1 Dr. Anders Ericsson
Department of Psychology
University of Colorado
Boulder, CO 80309

Private Sector

1 Mr. Wallace Feurzeig
Department of Educational Technology
Bolt Beranek & Newman
10 Moulton St.
Cambridge, MA 02238

1 Dr. Dexter Fletcher
University of Oregon
Department of Computer Science
Eugene, OR 97403

1 Dr. John R. Frederiksen
Bolt Beranek & Newman
50 Moulton Street
Cambridge, MA 02138

1 Dr. Michael Genesereth
Department of Computer Science
Stanford University
Stanford, CA 94305

1 Dr. Don Gentner
Center for Human Information Processing
University of California, San Diego
La Jolla, CA 92093

1 Dr. Dedre Gentner
Bolt Beranek & Newman
10 Moulton St.
Cambridge, MA 02138

1 Dr. Robert Glaser
Learning Research & Development Center
University of Pittsburgh
3939 O'Hara Street
PITTSBURGH, PA 15260

1 Dr. Joseph Goguen
SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025

1 Dr. Daniel Gopher
Faculty of Industrial Engineering
& Management
TECHNION
Haifa 32000
ISRAEL

1 DR. JAMES G. GREENO
LRDC
UNIVERSITY OF PITTSBURGH
3939 O'HARA STREET
PITTSBURGH, PA 15213

Private Sector

1 Dr. Barbara Hayes-Roth
Department of Computer Science
Stanford University
Stanford, CA 95305

1 Dr. Frederick Hayes-Roth
Teknowledge
525 University Ave.
Palo Alto, CA 94301

1 Dr. Joan I. Heller
Graduate Group in Science and
Mathematics Education
c/o School of Education
University of California
Berkeley, CA 94720

1 Dr. James R. Hoffman
Department of Psychology
University of Delaware
Newark, DE 19711

1 American Institutes for Research
1055 Thomas Jefferson St., N.W.
Washington, DC 20007

1 Dr. Kristina Hooper
Corporate Research, ATARI
1196 Borregas
Sunnyvale, CA 94086

1 Glenda Greenwald, Ed.
Human Intelligence Newsletter
P. O. Box 1163
Birmingham, MI 48012

1 Dr. Earl Hunt
Dept. of Psychology
University of Washington
Seattle, WA 98105

1 Robin Jeffries
Computer Research Center
Hewlett-Packard Laboratories
1501 Page Mill Road
Palo Alto, CA 94304

1 Dr. Marcel Just
Department of Psychology
Carnegie-Mellon University
Pittsburgh, PA 15213

ONR Distribution List

Private Sector

1 Dr. David Kieras
Department of Psychology
University of Arizona
Tucson, AZ 85721

1 Dr. Stephen Kosslyn
1236 William James Hall
33 Kirkland St.
Cambridge, MA 02138

1 Dr. Pat Langley
The Robotics Institute
Carnegie-Mellon University
Pittsburgh, PA 15213

1 Dr. Jill Larkin
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

1 Dr. Alan Lesgold
Learning R&D Center
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 15260

1 Dr. Jim Levin
University of California
at San Diego
Laboratory for Comparative
Human Cognition - D003A
La Jolla, CA 92093

1 Dr. Marcia C. Linn
Lawrence Hall of Science
University of California
Berkeley, CA 94720

1 Dr. Don Lyon
P. O. Box 44
Higley , AZ 85236

1 Dr. Jay McClelland
Department of Psychology
MIT
Cambridge, MA 02139

1 Dr. James R. Miller
Computer•Thought Corporation
1721 West Plano Highway
Plano, TX 75075



Private Sector

1 Dr. Mark Miller
Computer•Thought Corporation
1721 West Plano Parkway
Plano, TX 75075

1 Dr. Tom Moran
Xerox PARC
3333 Coyote Hill Road
Palo Alto, CA 94304

1 Dr. Allen Munro
Behavioral Technology Laboratories
1845 Elena Ave., Fourth Floor
Redondo Beach, CA 90277

1 Dr. Jesse Orlansky
Institute for Defense Analyses
1801 N. Beauregard St.
Alexandria, VA 22311

1 Dr. Nancy Pennington
University of Chicago
Graduate School of Business
1101 E. 58th St.
Chicago, IL 60637

1 DR. PETER POLSON
DEPT. OF PSYCHOLOGY
UNIVERSITY OF COLORADO
BOULDER, CO 80309

1 Dr. Lynn Reder
Department of Psychology
Carnegie-Mellon University
Schenley Park
Pittsburgh, PA 15213

1 Dr. Fred Reif
Physics Department
University of California
Berkeley, CA 94720

1 Dr. Lauren Resnick
LRDC
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 1521

1 Dr. Jeff Richardson
Denver Research Institute
University of Denver
Denver, CO 80208

Private Sector

1 Mary S. Riley
Program in Cognitive Science
Center for Human Information Processing
University of California, San Diego
La Jolla, CA 92093

1 Dr. Andrew M. Rose
American Institutes for Research
1055 Thomas Jefferson St. NW
Washington, DC 20007

1 Dr. Ernst Z. Rothkopf
Bell Laboratories
Murray Hill, NJ 07974

1 Dr. William B. Rouse
Georgia Institute of Technology
School of Industrial & Systems
Engineering
Atlanta, GA 30332

1 Dr. David Rumelhart
Center for Human Information Processing
Univ. of California, San Diego
La Jolla, CA 92093

1 Dr. Michael J. Saxon
Perceptronics, Inc
6271 Variel Avenue
Woodland Hills, CA 91364

1 Dr. Roger Schank
Yale University
Department of Computer Science
P.O. Box 2158
New Haven, CT 06520

1 Dr. Walter Schneider
Psychology Department
603 E. Daniel
Champaign, IL 61820

1 Dr. Alan Schoenfeld
Mathematics and Education
The University of Rochester
Rochester, NY 14627

1 Mr. Colin Sheppard
Applied Psychology Unit
Admiralty Marine Technology Est.
Teddington, Middlesex
United Kingdom

Private Sector

1 Dr. H. Wallace Sinaiko
Program Director
Manpower Research and Advisory Services
Smithsonian Institution
801 North Pitt Street
Alexandria, VA 22314

1 Dr. Edward E. Smith
Bolt Beranek & Newman, Inc.
50 Moulton Street
Cambridge, MA 02138

1 Dr. Elliott Soloway
Yale University
Department of Computer Science
P.O. Box 2158
New Haven, CT 06520

1 Dr. Kathryn T. Spoehr
Psychology Department
Brown University
Providence, RI 02912

1 Dr. Robert Sternberg
Dept. of Psychology
Yale University
Box 11A, Yale Station
New Haven, CT 06520

1 Dr. Albert Stevens
Bolt Beranek & Newman, Inc.
10 Moulton St.
Cambridge, MA 02238

1 Dr. David Stone
KAJ Software, Inc.
3420 East Shea Blvd.
Suite 161
Phoenix, AZ 85028

1 Dr. Perry W. Thorndyke
Perceptronics, Inc.
545 Middlefield Road, Suite 140
Menlo Park, CA 94025

1 Dr. Douglas Towne
Univ. of So. California
Behavioral Technology Labs
1845 S. Elena Ave.
Redondo Beach, CA 90277

ONR Distribution List

Private Sector

1 Dr. Kurt Van Lehn
Xerox PARC
3333 Coyote Hill Road
Palo Alto, CA 94304

1 Dr. Keith T. Wescourt
Perceptronics, Inc.
545 Middlefield Road, Suite 140
Menlo Park, CA 94025

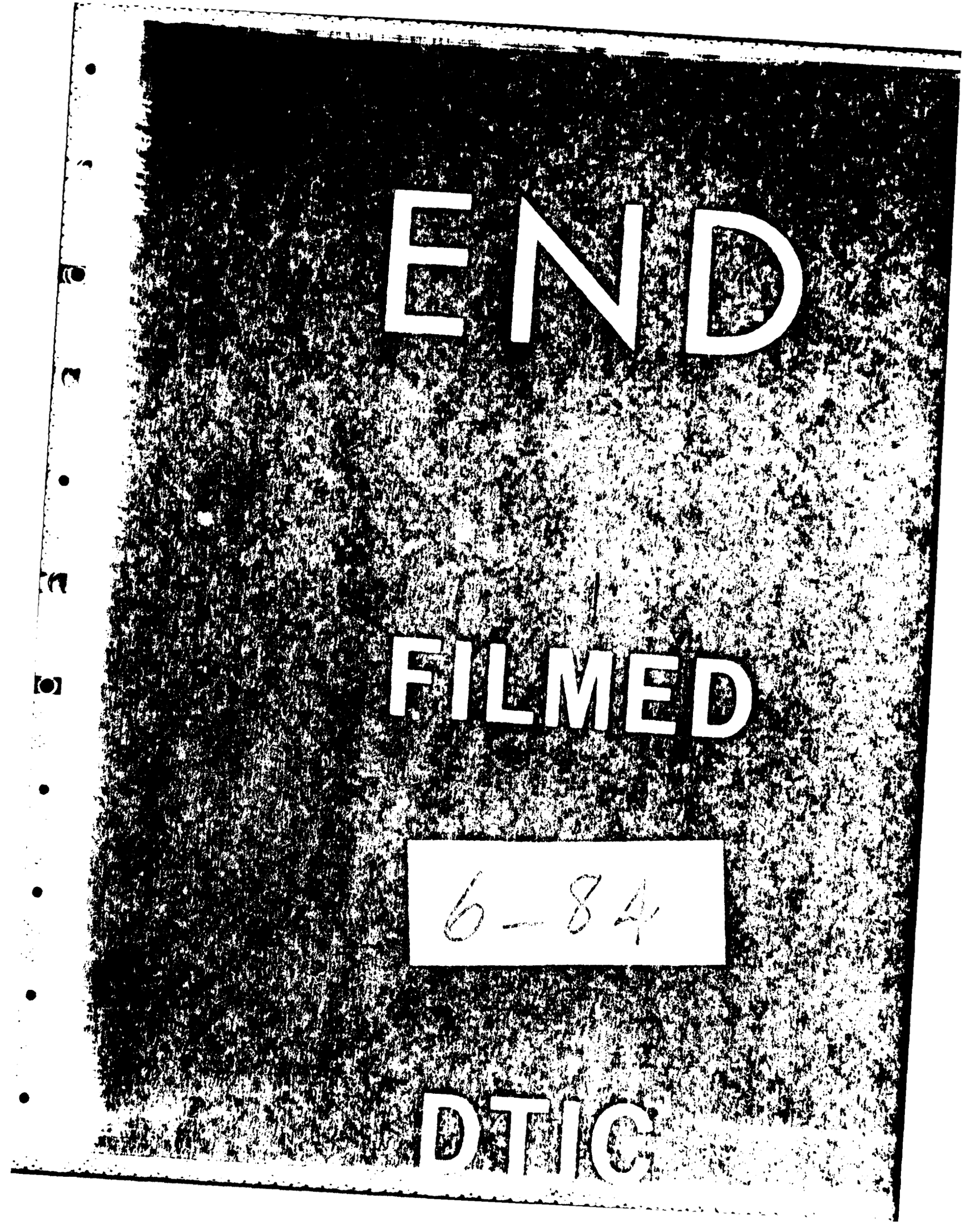
1 William B. Whitten
Bell Laboratories
2B-410
Holmdel, NJ 07733

1 Dr. Thomas Wickens
Department of Psychology
Franz Hall
University of California
405 Hilgards Avenue
Los Angeles, CA 90024

1 Dr. Mike Williams
IntelliGenetics
124 University Avenue
Palo Alto, CA 94301

1 Dr. Joseph Wohl
Alphatech, Inc.
2 Burlington Executive Center
111 Middlesex Turnpike
Burlington, MA 01803

ONR Distribution List


END
FILMED
6-84
DTIC